<EURO/SYS'20>

AlloX: Compute Allocation in Hybrid Clusters

Tan N. Le Xiao Sun Mosharaf Chowdhury Zhenhua Liu

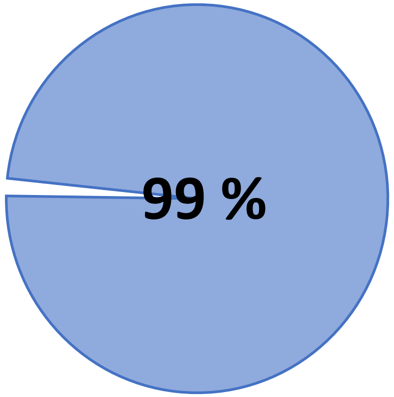
Stony Brook University SUNY Korea The State University of New York University of Michigan

tnle@cs.stonybrook.edu

Resource Allocation in Clusters

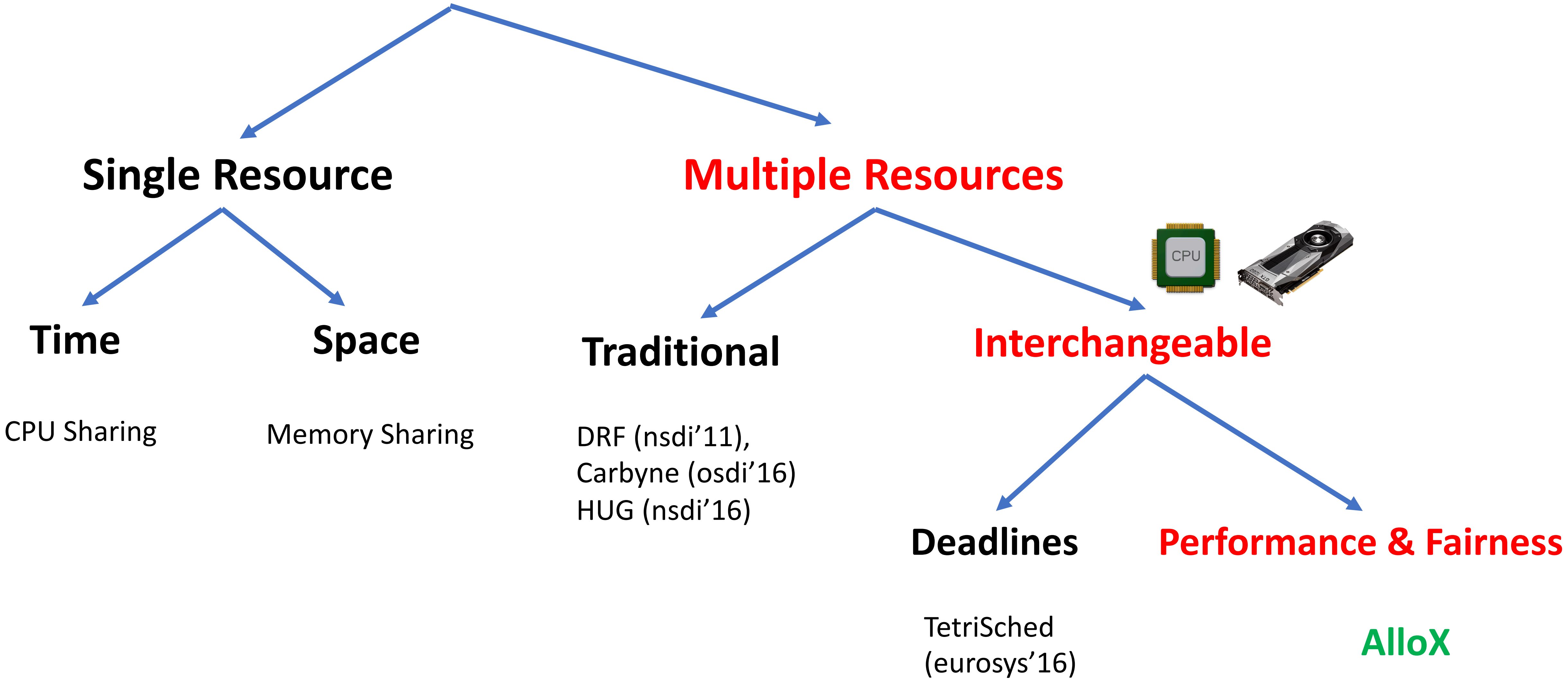
Performance

Fairness



Utilization

Resource Allocation Design Space



Single Resource

Multiple Resources

Time

CPU Sharing

Space

Memory Sharing

Traditional

DRF (nsdi'11),
Carbyne (osdi'16)
HUG (nsdi'16)

Interchangeable

Deadlines

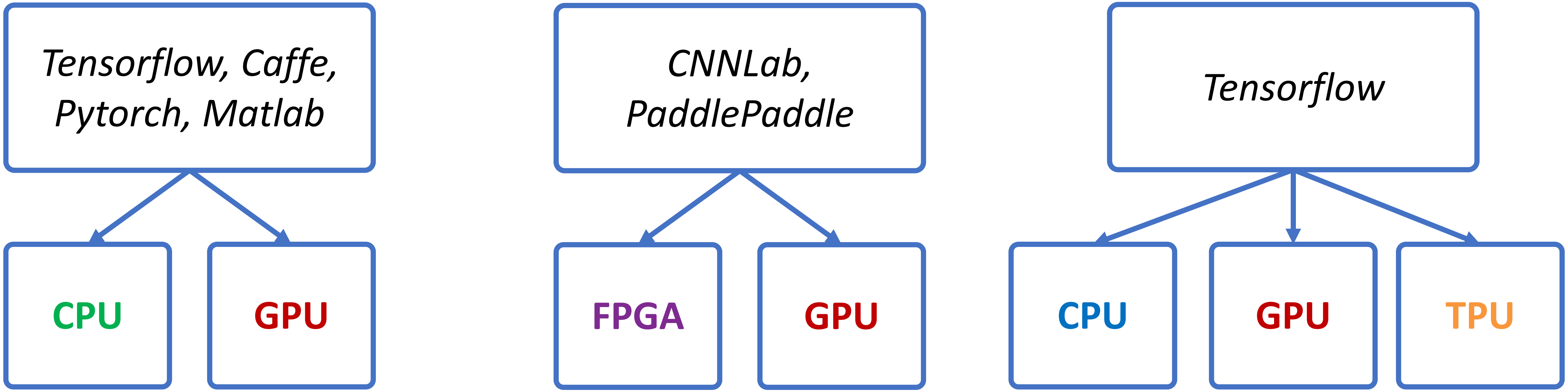
TetriSched
(eurosys'16)

Performance & Fairness

AlloX

Interchangeability in Resources

Same applications run on different resource types

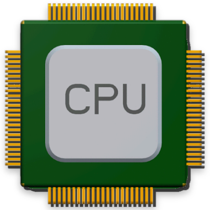


Modern Frameworks support Interchangeability

https://github.com/PaddlePaddle/Paddle
https://github.com/cnnlabs

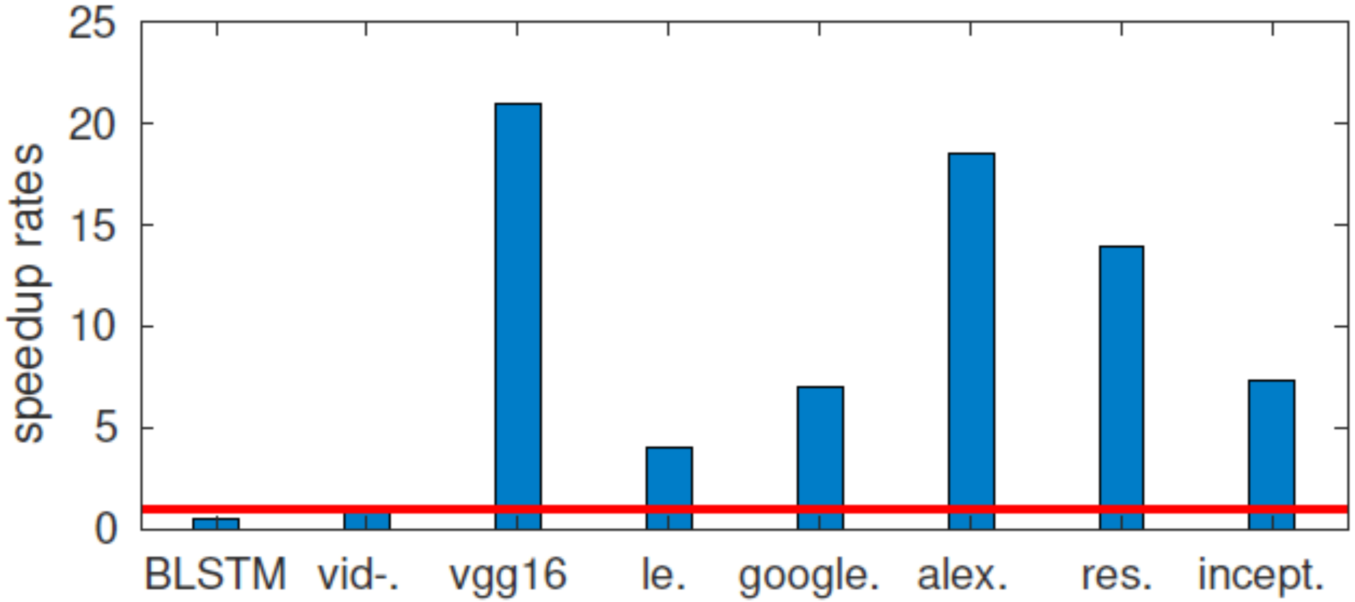
Heterogeneity in hybrid CPU/GPU Clusters

Traditional nodes



Expensive GPUs

Speed-up rates are distinct

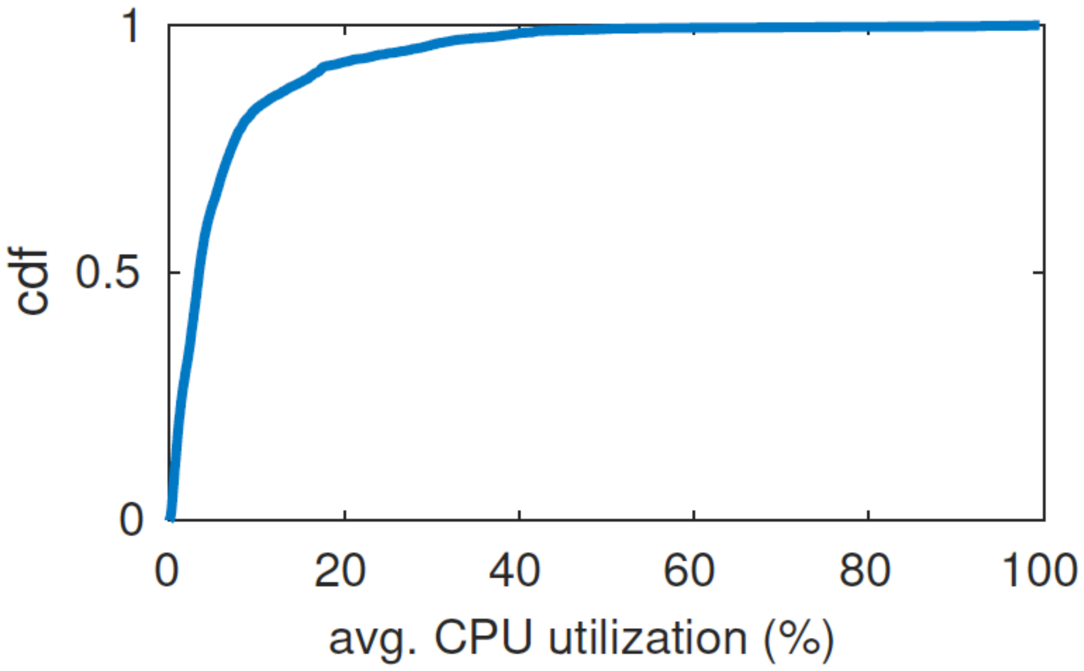


Intel E5 2.4Ghz CPU vs. Nvidia K80 GPU

Overload if most users prefer GPUs

Expensive GPUs are overloaded while **CPUs are under-utilized**

Microsoft Azure

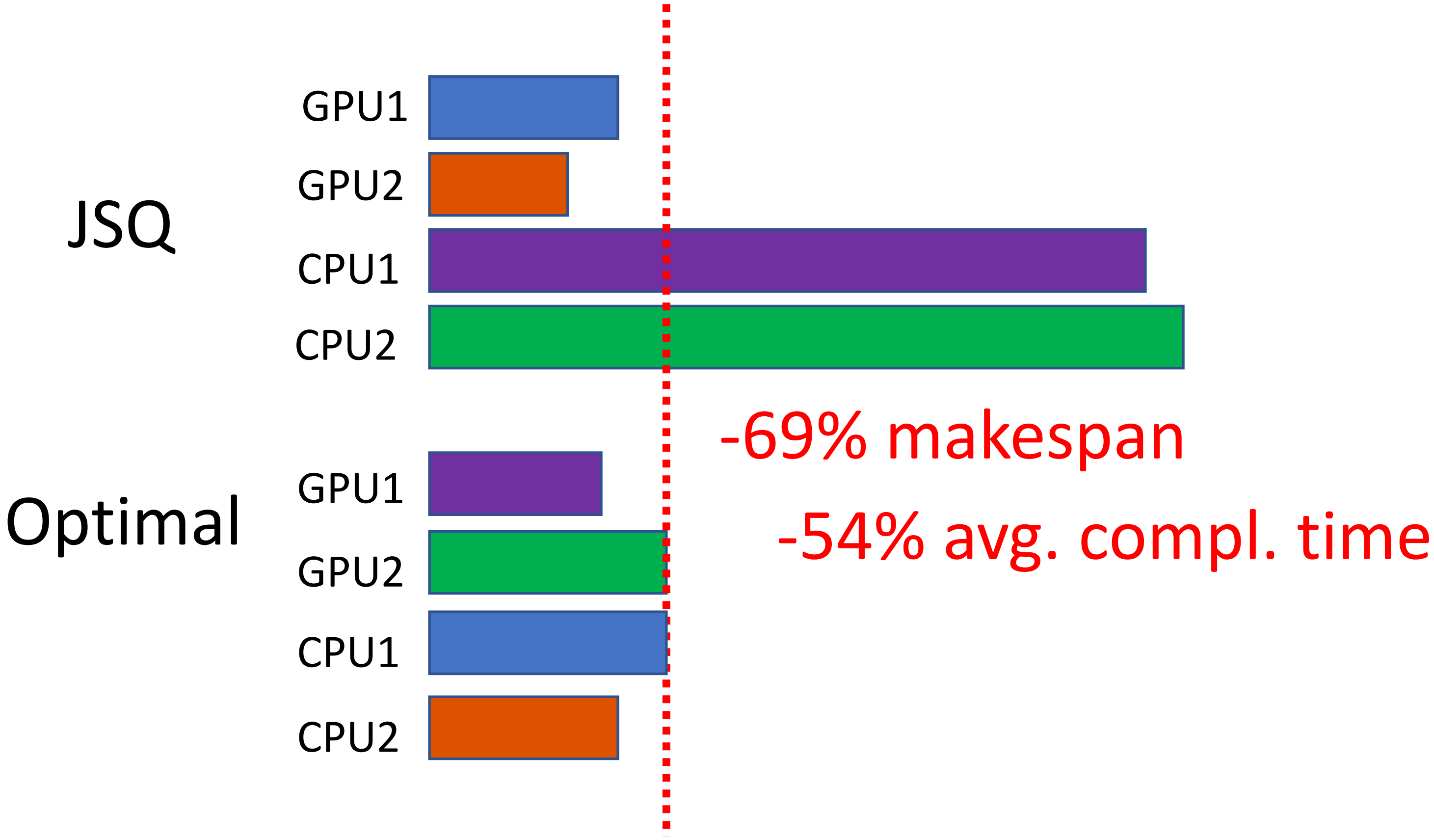


Let’s explore some solutions

Join the Shortest Queue (JSQ)

Processing times
(GPU, CPU)

J1	(**40**, 50)
J2	(**30**, 40)
J3	(35, **150**)
J4	(50, **160**)



-69% makespan

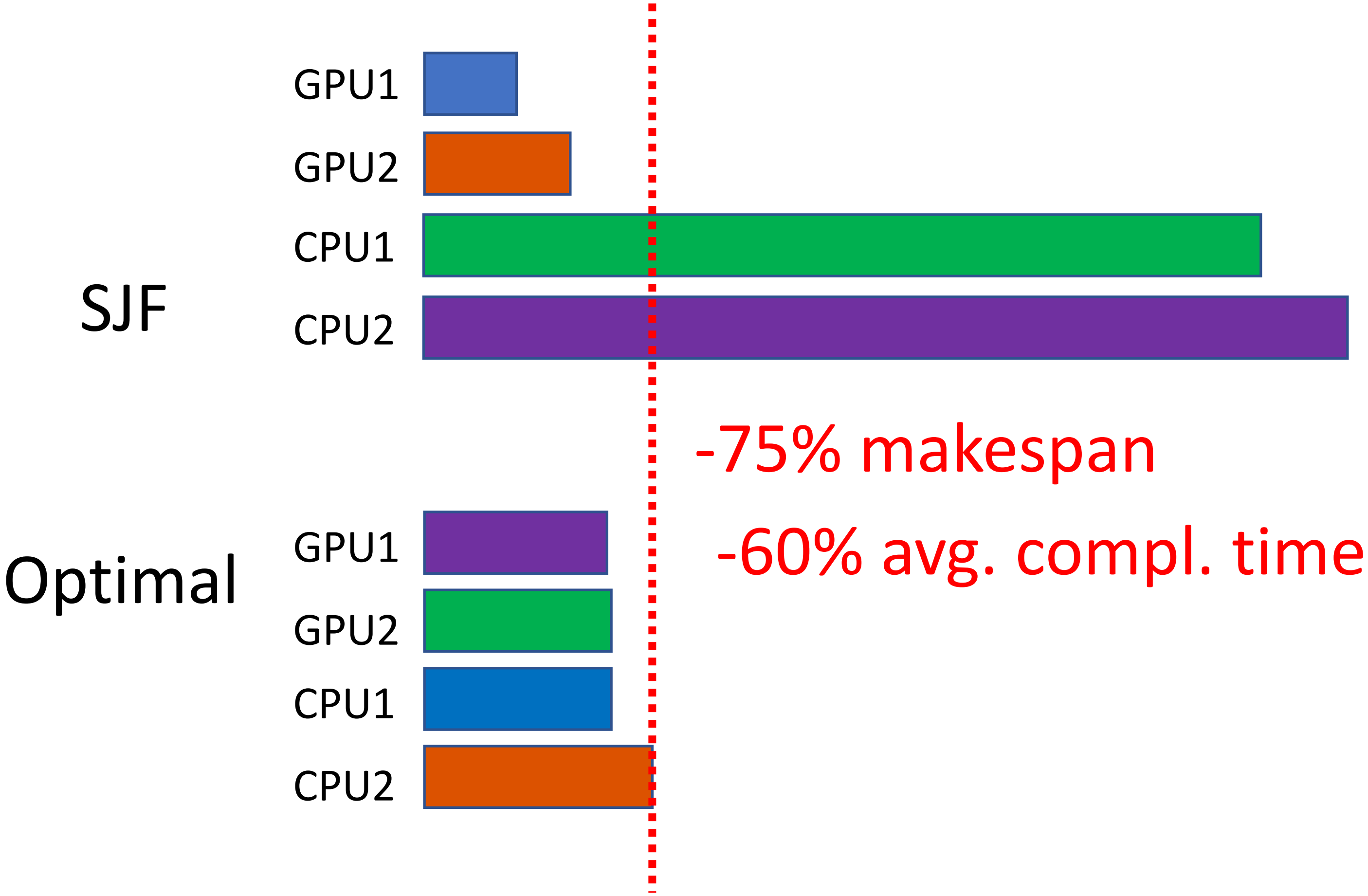
-54% avg. compl. time

JSQ does not consider processing times

Shortest Job First (SJF)

Processing times
(GPU, CPU)

J1	(**10**, 20)
J2	(**15**, 25)
J3	(20, **100**)
J4	(20, **90**)



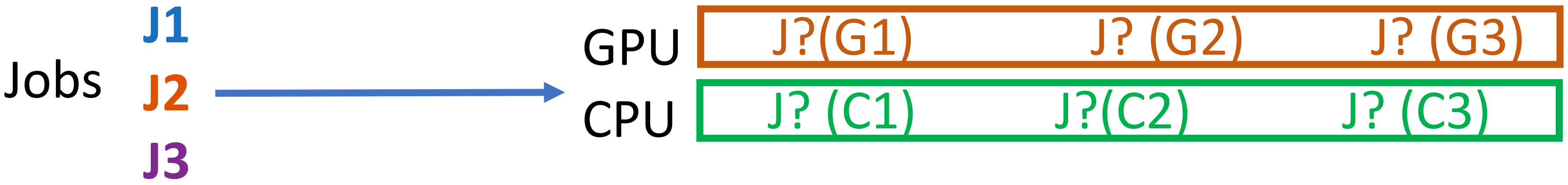
-75% makespan

-60% avg. compl. time

SJF does not consider speed-up rates

AlloX – Minimize Avg. Completion Time

Convert the scheduling & placement



Jobs

J1

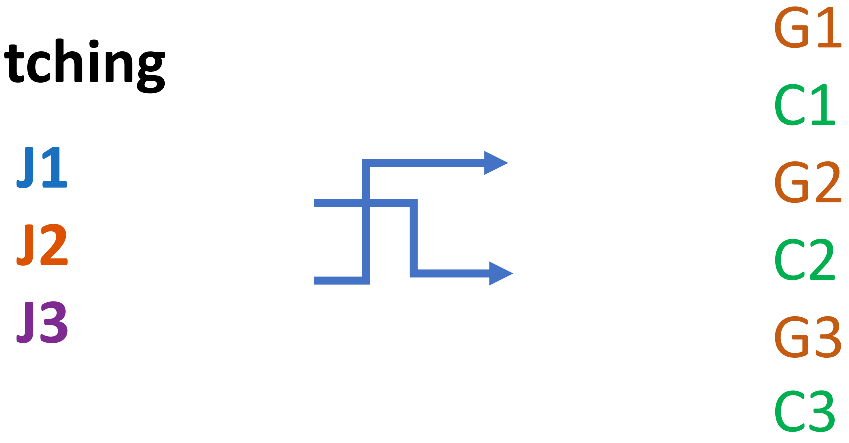
J2

J3

GPU J?(G1) J? (G2) J? (G3)

CPU J? (C1) J?(C2) J? (C3)

into min-cost bipartite matching



J1

J2

J3

G1

C1

G2

C2

G3

C3

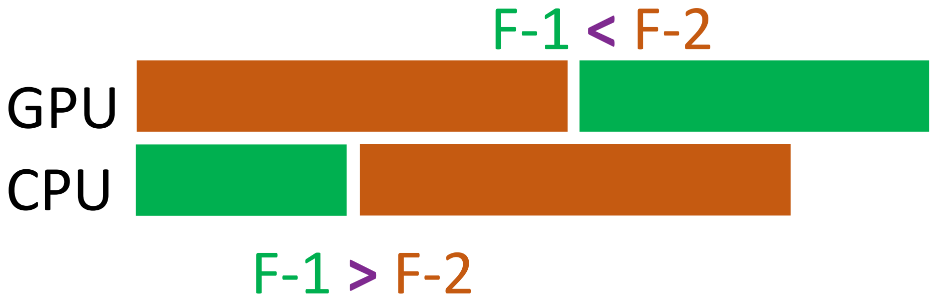
solved in polynomial time

AlloX – Maintains Fairness for interchangeable resources

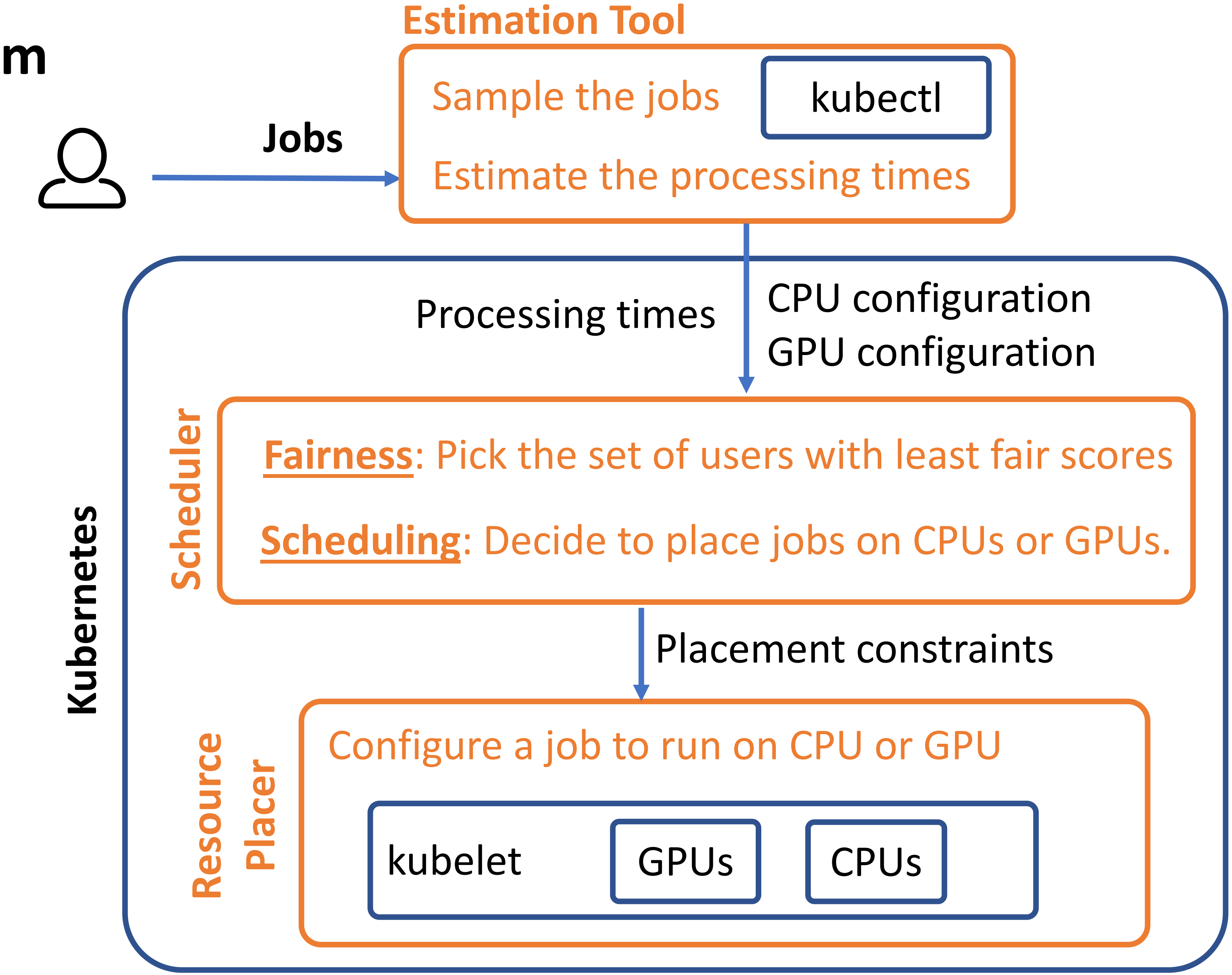
User A may not be happy if we keep putting him on CPU.

Idea: Prioritize users with low fairness scores *F* who run jobs on the unfavorable resources

User 1	F-1
User 2	F-2



AlloX System



Estimation Tool
Sample the jobs
kubectl
Estimate the processing times
Jobs
Processing times
CPU configuration
GPU configuration
Kubernetes
Scheduler
Fairness: Pick the set of users with least fair scores
Scheduling: Decide to place jobs on CPUs or GPUs.
Placement constraints
Resource Placer
Configure a job to run on CPU or GPU
kubelet
GPUs
CPUs

Performance of AlloX

DRF: Dominant Resource Fairness + FIFO

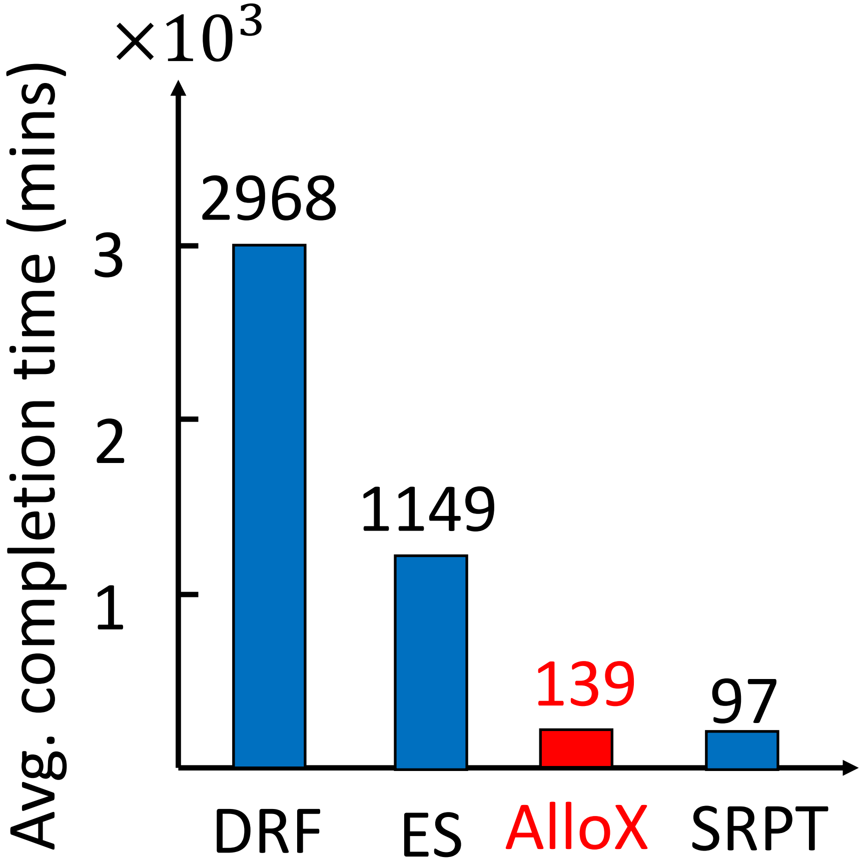
Resource configurations are fixed

ES: Equal Share + SJF

Keep filling the available resources

SRPT: Shortest Remaining Processing Time

Impractical switching between CPU&GPU



AlloX reduces up to 95% avg. completion time

TensorFlow CNN benchmarks

<EURO/SYS’20>

AlloX: Compute Allocation in Hybrid Clusters

Tan N. Le   Xiao Sun   Mosharaf Chowdhury   Zhenhua Liu

Stony Brook University   SUNY Korea The State University of New York   UNIVERSITY OF MICHIGAN

tnle@cs.stonybrook.edu